Training an Adaptive Critic Flight Controller

Silvia Ferrari

Advisor: Prof. Robert F. Stengel

Princeton University

Introduction

- **Classical/neural synthesis** of control systems
   Prior knowledge
   Adaptive control and artificial neural networks

- **Adaptive critics**
   Learn in real time
   Cope with noise
   Cope with many variables
   Plan over time in a complex way

   ...

- Adaptation takes place during every time interval:

   Action network takes immediate control action

   Critic network estimates projected cost

Motivation

- Provide full envelope control

- Multiphase learning:
  Pre-training phase, motivated by corresponding linear controller
  On-line training phase, during simulations or testing

- On-line training accounts for:
  Differences between actual and assumed dynamic models
  Nonlinear effects not captured in linearizations

- Potential applications:
  Incorporate pilot's knowledge into controller *a-priori*
  Uninhabited air vehicles control
  Aerobatic flight control

Table of Contents

- Aircraft control design approach

- Initialization, or pre-training phase

- Adaptive critic neural network controller
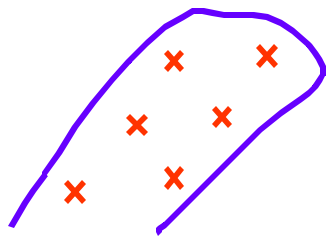
- On-line training

- Resilient backpropagation

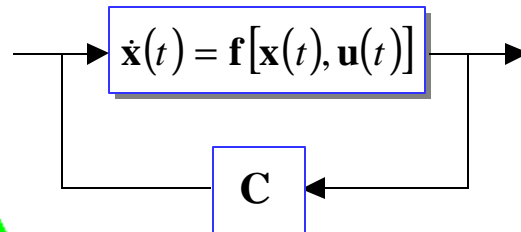# Aircraft Control Design Approach

Modeling

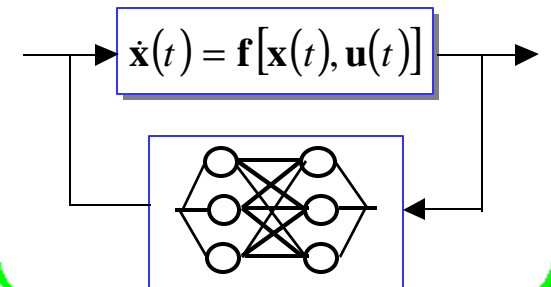$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t)]$$

Linearizations

Linear Control
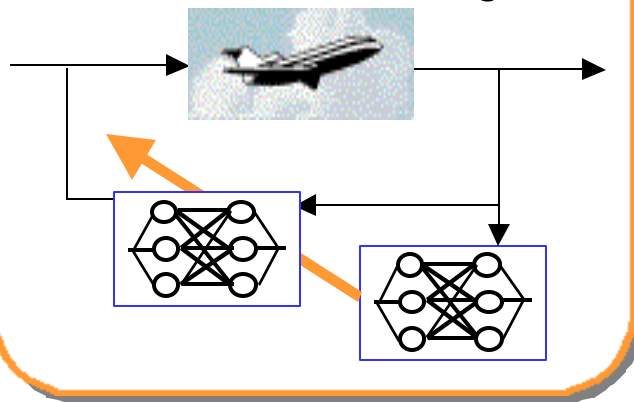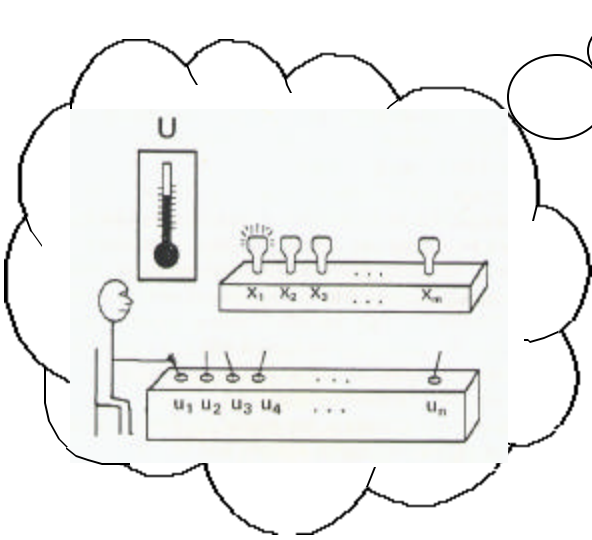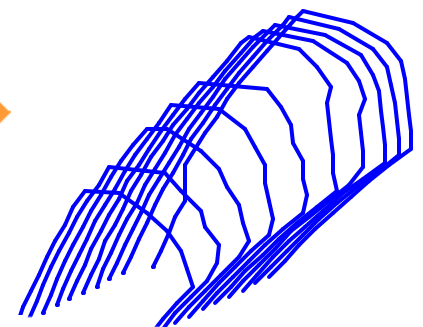
$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t)]$$

C

Initialization

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t)]$$

On-line Training

U

x_1 x_2 x_3 $\ldots$ x_n

u_1 u_2 u_3 u_4 $\ldots$ u_n

Full Envelope Control!

Linear Control Design

Linearizations:

$$\dot{\mathbf{x}}(t) = \mathbf{f}\left[\mathbf{x}(t), \mathbf{u}(t), \mathbf{p}(t)\right]$$

$$\Downarrow$$

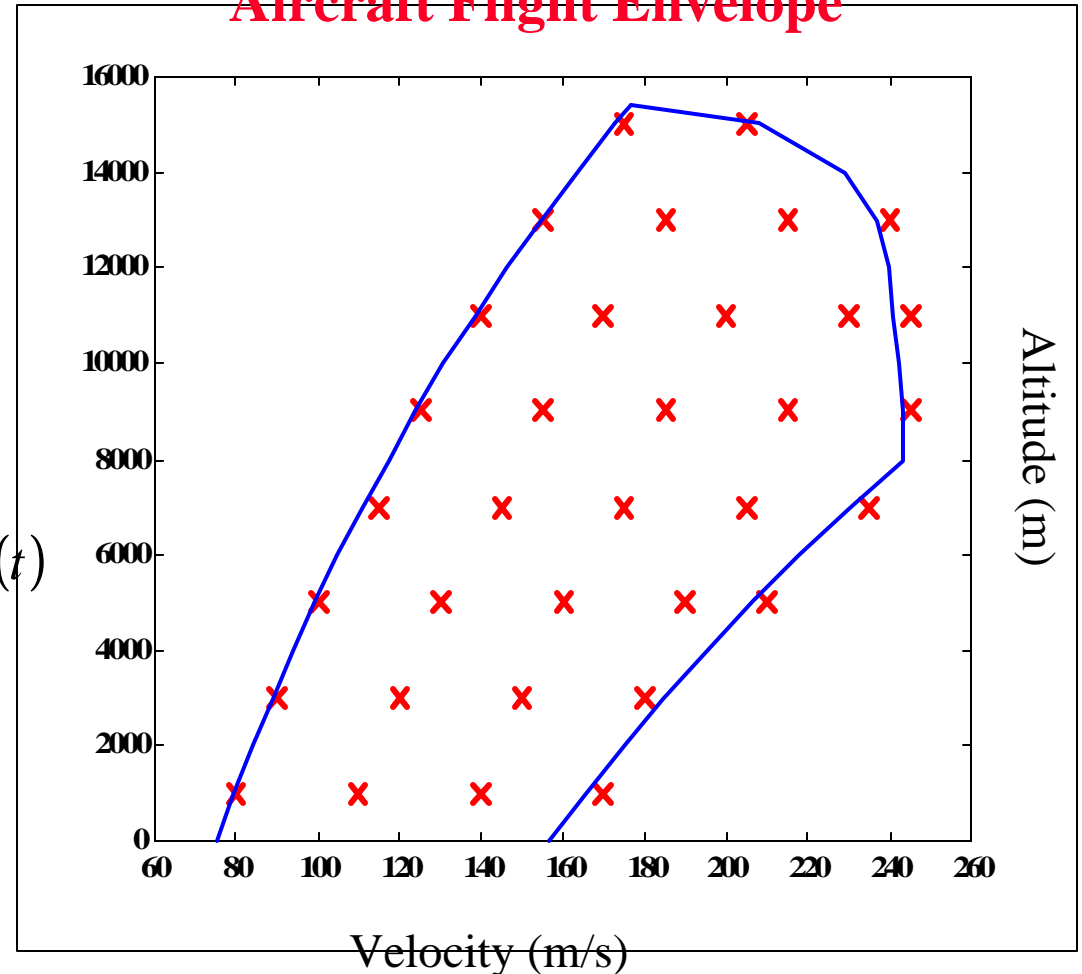$$\Delta\dot{\mathbf{x}}(t) = \mathbf{F}\Delta\mathbf{x}(t) + \mathbf{G}\Delta\mathbf{u}(t)$$

$$\Downarrow$$

$$\begin{cases} \Delta\dot{\mathbf{x}}_L(t) = \mathbf{F}_L\Delta\mathbf{x}_L(t) + \mathbf{G}_L\Delta\mathbf{u}_L(t) \\ \Delta\dot{\mathbf{x}}_{LD}(t) = \mathbf{F}_{LD}\Delta\mathbf{x}_{LD}(t) + \mathbf{G}_{LD}\Delta\mathbf{u}_{LD}(t) \end{cases}$$
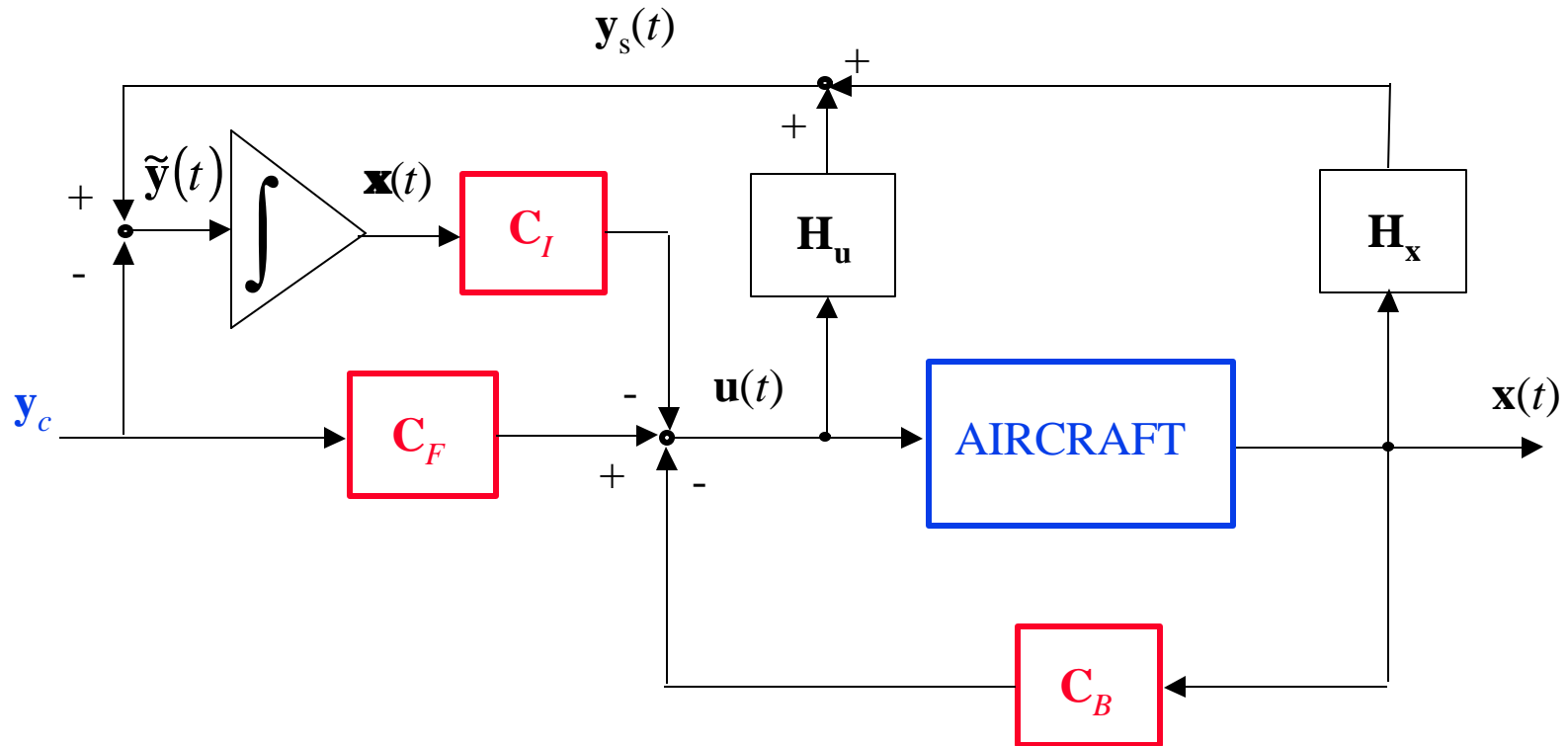
Linear control design:

- Longitudinal
- Lateral-directional

Aircraft Flight Envelope

Velocity (m/s)

Altitude (m)

Linear Proportional-Integral Controller

Closed-loop stability: $\mathbf{x}(t) \to \mathbf{x}_c, \ \ \mathbf{u}(t) \to \mathbf{u}_c, \ \ \tilde{\mathbf{y}}(t) \to 0$

Omitting Δ's, for simplicity:

$\tilde{\mathbf{y}}(t) = \mathbf{y}_S(t) - \mathbf{y}_C, \ \ \tilde{\mathbf{u}}(t) = \mathbf{u}(t) - \mathbf{u}_C, \ldots, \ \ \ \mathbf{y}_c = \text{desired output}, \ (\mathbf{x}_c, \mathbf{u}_c) = \text{set point}.$
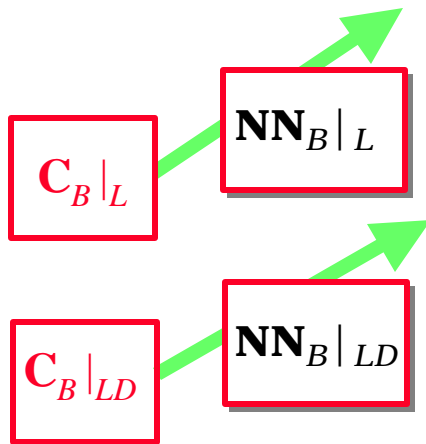
Proportional-Integral Neural Network Controller
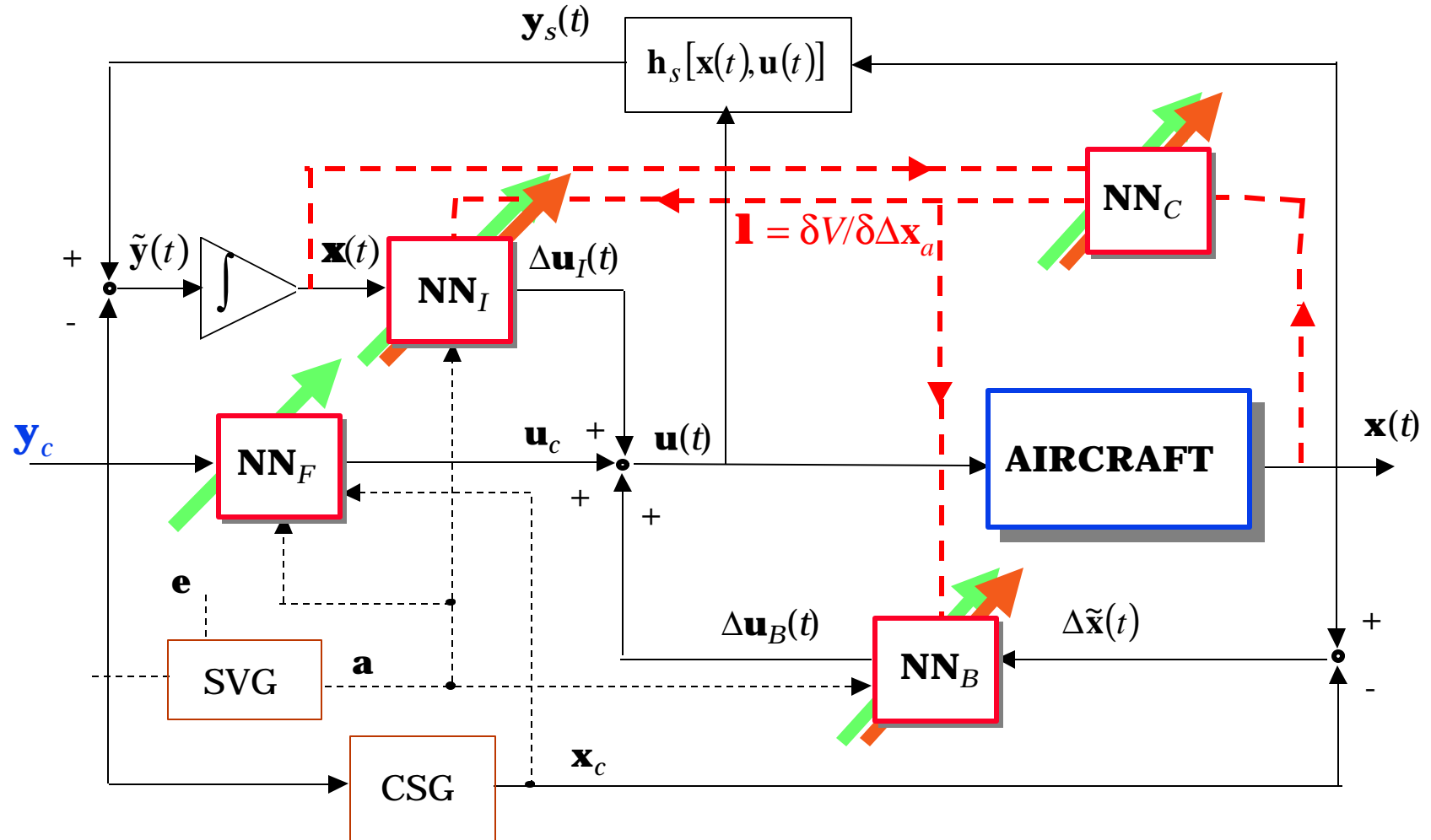
Where: $\quad \mathbf{x}(t) \rightarrow \mathbf{x}_c, \quad \mathbf{u}(t) \rightarrow \mathbf{u}_c, \quad \tilde{\mathbf{y}}(t) \rightarrow 0, \quad \mathbf{y}_s(t) \rightarrow \mathbf{y}_c$

Algebraic Neural Network Pre-training Phase

$\mathbf{C}_B|_L$ $\mathbf{NN}_B|_L$

$\mathbf{C}_B|_{LD}$ $\mathbf{NN}_B|_{LD}$

Integral Error:

$\mathbf{C}_I|_L$ $\mathbf{NN}_I|_L$

$\mathbf{C}_I|_{LD}$ $\mathbf{NN}_I|_{LD}$

Critic:

$\mathbf{P}|_L$ $\mathbf{NN}_C|_L$

$\mathbf{P}|_{LD}$ $\mathbf{NN}_C|_{LD}$
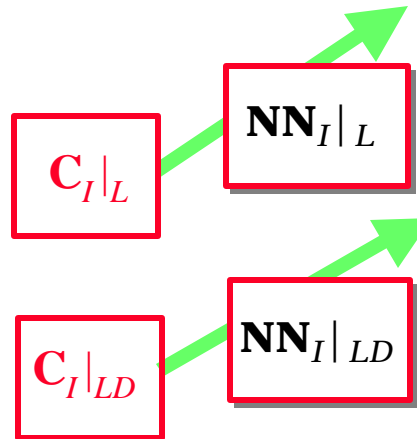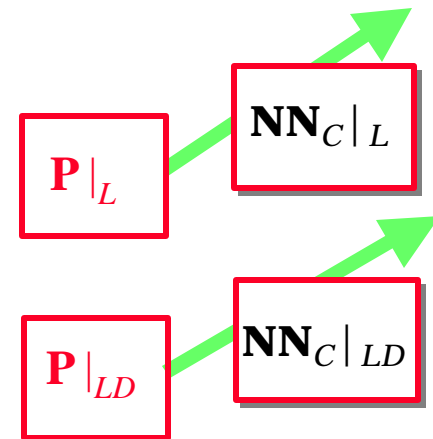
Combine longitudinal and
lateral-directional networks:

$\mathbf{NN}_B|_L$

$\mathbf{NN}_B|_{LD}$ $\Rightarrow$ $\mathbf{NN}_B$, etc. ...
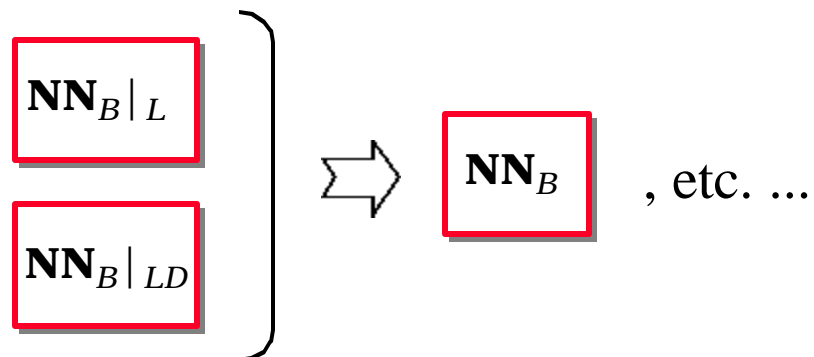
Obtain action network:

$\mathbf{NN}_B$

$\mathbf{NN}_I$ $\Rightarrow$ $\mathbf{NN}_A$
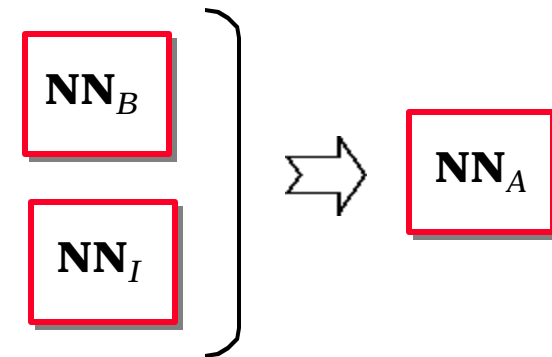
Comparison of Neural Network and Linear Controllers Between Training Points
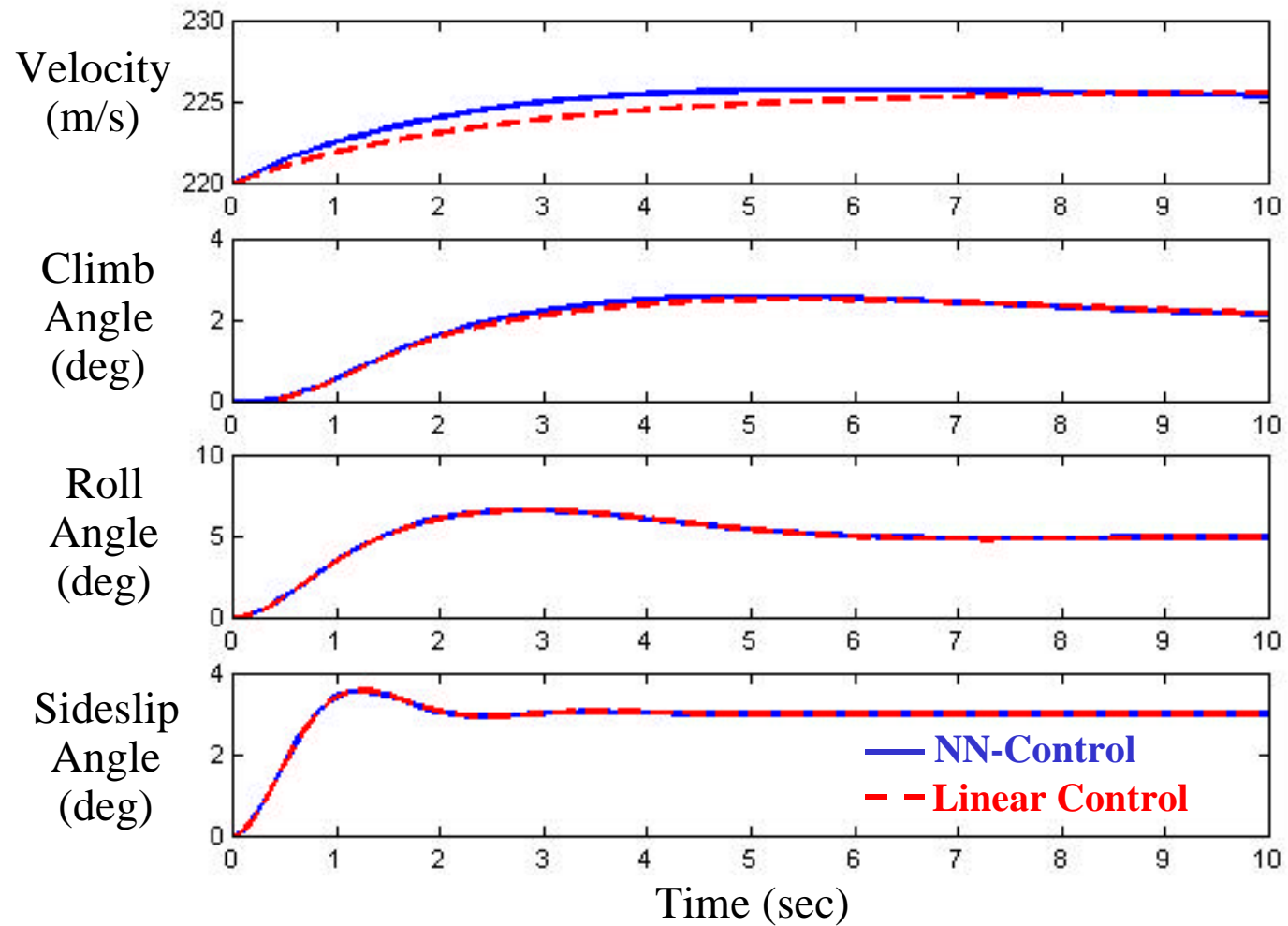
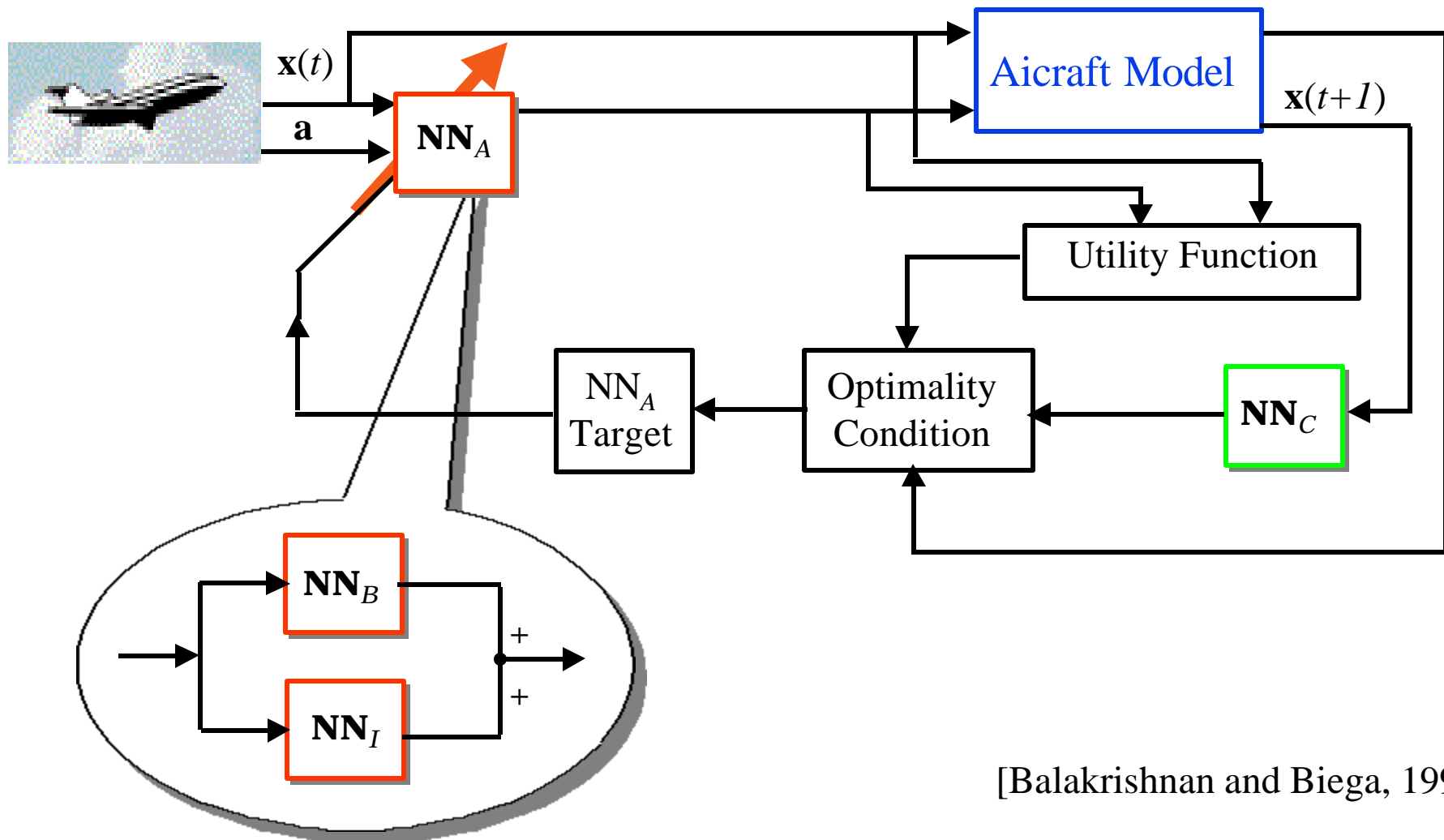Flight condition: (14,000 m; 220 m/s)

Velocity and Climb Angle Command

Roll and Sideslip Angle Command

Adaptive Critic Implementation:
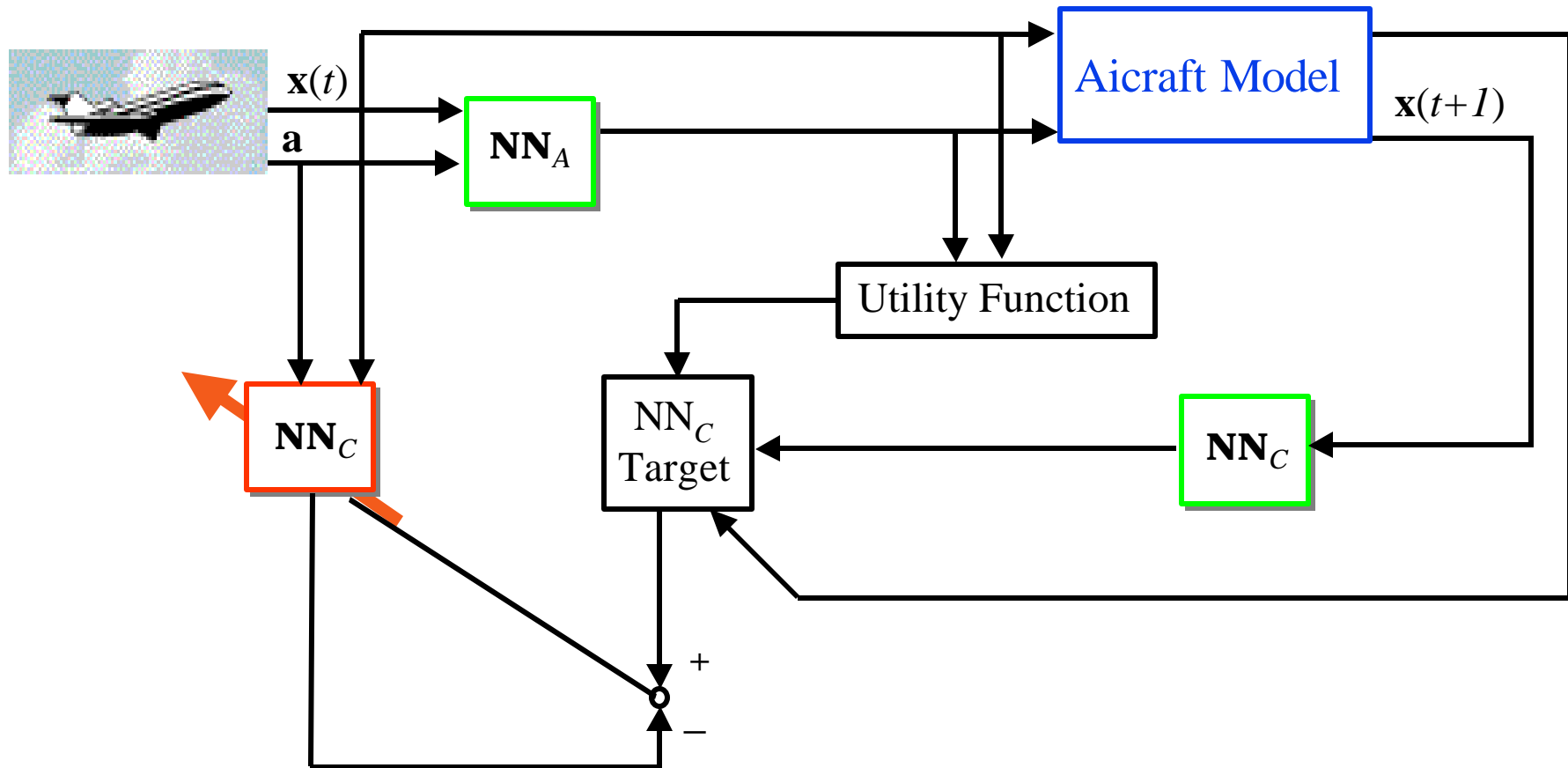# Action Network On-line Training

Train action network, at time t, holding the critic parameters fixed

[Balakrishnan and Biega, 1996]

Adaptive Critic Implementation:
Critic Network On-line Training

Train critic network, at time t, holding the action parameters fixed

[Balakrishnan and Biega, 1996]

On-line Neural Network Training Goal

- Given a target, $\mathbf{t}(\mathbf{p})$, for the network output, $\mathbf{z}(\mathbf{p})$:

$$\min_{\mathbf{w}} E \equiv \min_{\mathbf{w}} \left\{ \| \mathbf{t}(\mathbf{p}) - \mathbf{z}(\mathbf{p}) \|^2 \right\} \quad \begin{cases} \mathbf{p} \equiv \text{network input} \\ \\ E \equiv \text{network performance} \end{cases}$$

  with network parameters, $\mathbf{w}$, provided by the initialization phase.

Scaling effect:
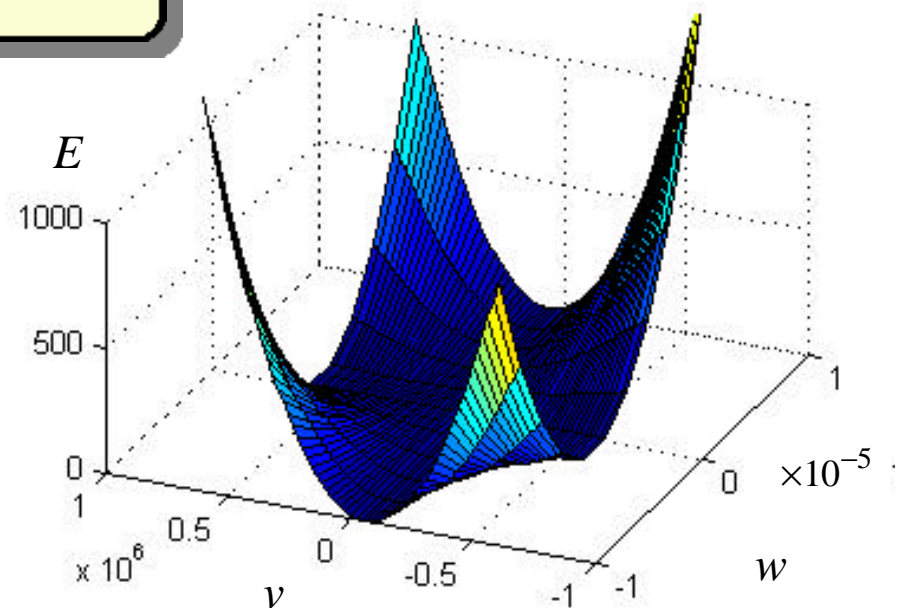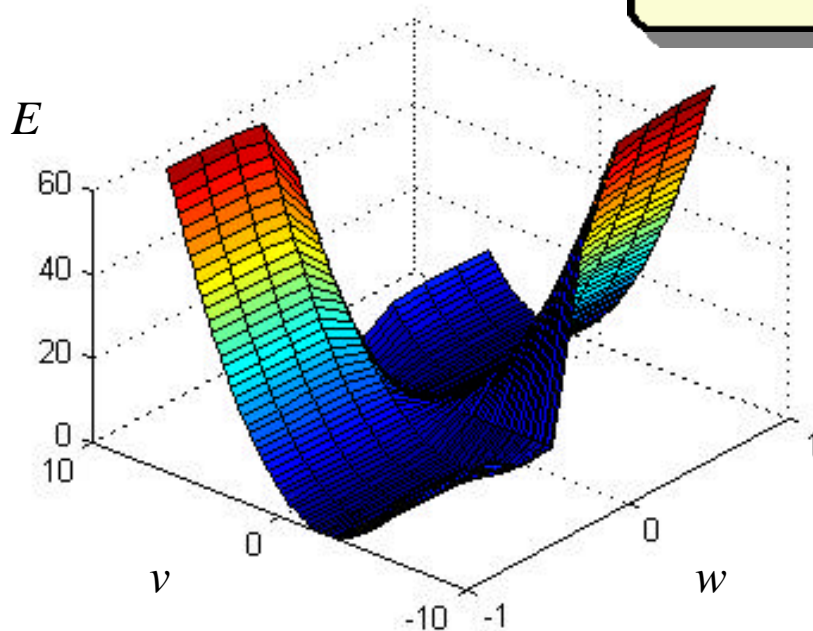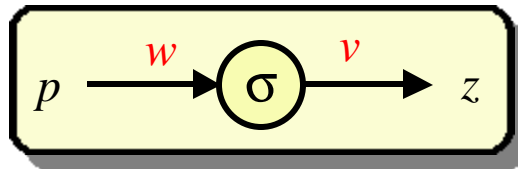
Comparison of Neural Network Training Algorithms

Technique	Speed	Implement. Complexity	Memory Requirement	Main Drawbacks
Backpropagation	Poor	Low	Small	• Scaling • Speed
Levenberg-Marquardt	Excellent	Medium	Large	• Memory • Complexity
Extended Kalman Filter	Excellent (Highest)	High	Large	• Memory • Complexity
Resilient Backpropagation	Medium-High	Low	Medium-Small	• Local convergence

Resilient Backpropagation

NN Architecture and $\mathbf{w}$ (initialization)

$$\Delta^k = \begin{cases} \mathbf{h}^+\Delta^{(k-1)}, & \text{if } \dfrac{\partial E^{(k-1)}}{\partial w}\dfrac{\partial E^k}{\partial w} > 0 \\[2ex] \mathbf{h}^-\Delta^{(k-1)}, & \text{if } \dfrac{\partial E^{(k-1)}}{\partial w}\dfrac{\partial E^k}{\partial w} < 0 \\[2ex] \Delta^{(k-1)}, & \text{else} \end{cases}$$

$$\Delta w^k = \begin{cases} -\Delta^k, & \text{if } \dfrac{\partial E^k}{\partial w} > 0 \\[2ex] +\Delta^k, & \text{if } \dfrac{\partial E^k}{\partial w} < 0 \\[2ex] 0, & \text{else} \end{cases}$$

$$\Delta w^k = -\Delta w^{(k-1)}$$

$$w^{(k+1)} = w^k + \Delta w^k$$

Store $\mathbf{w}$, and $\mathbf{D}$
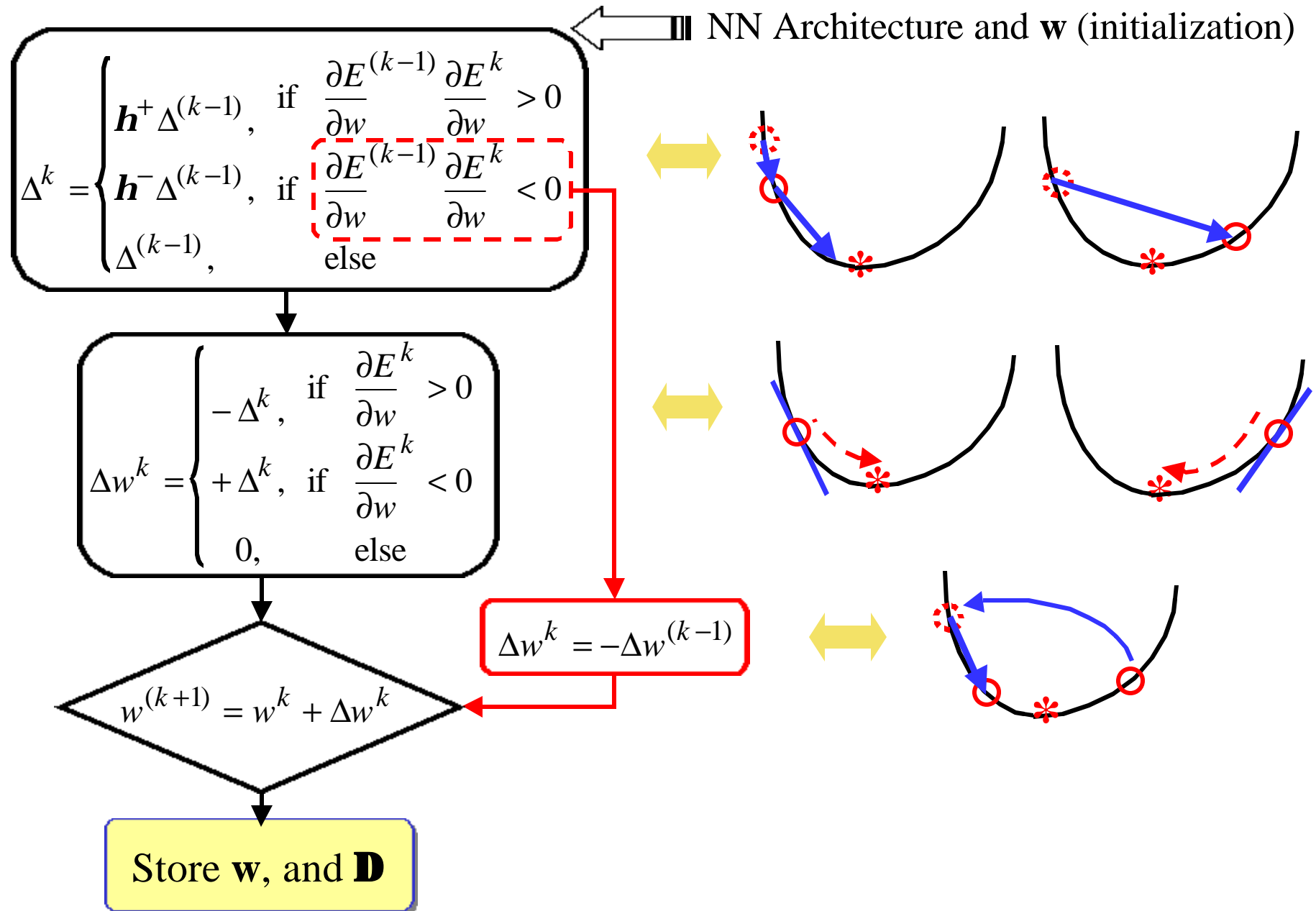
Resilient Backpropagation Algorithm Performance

Adaptive critics neural network controller test case: **Action Network**
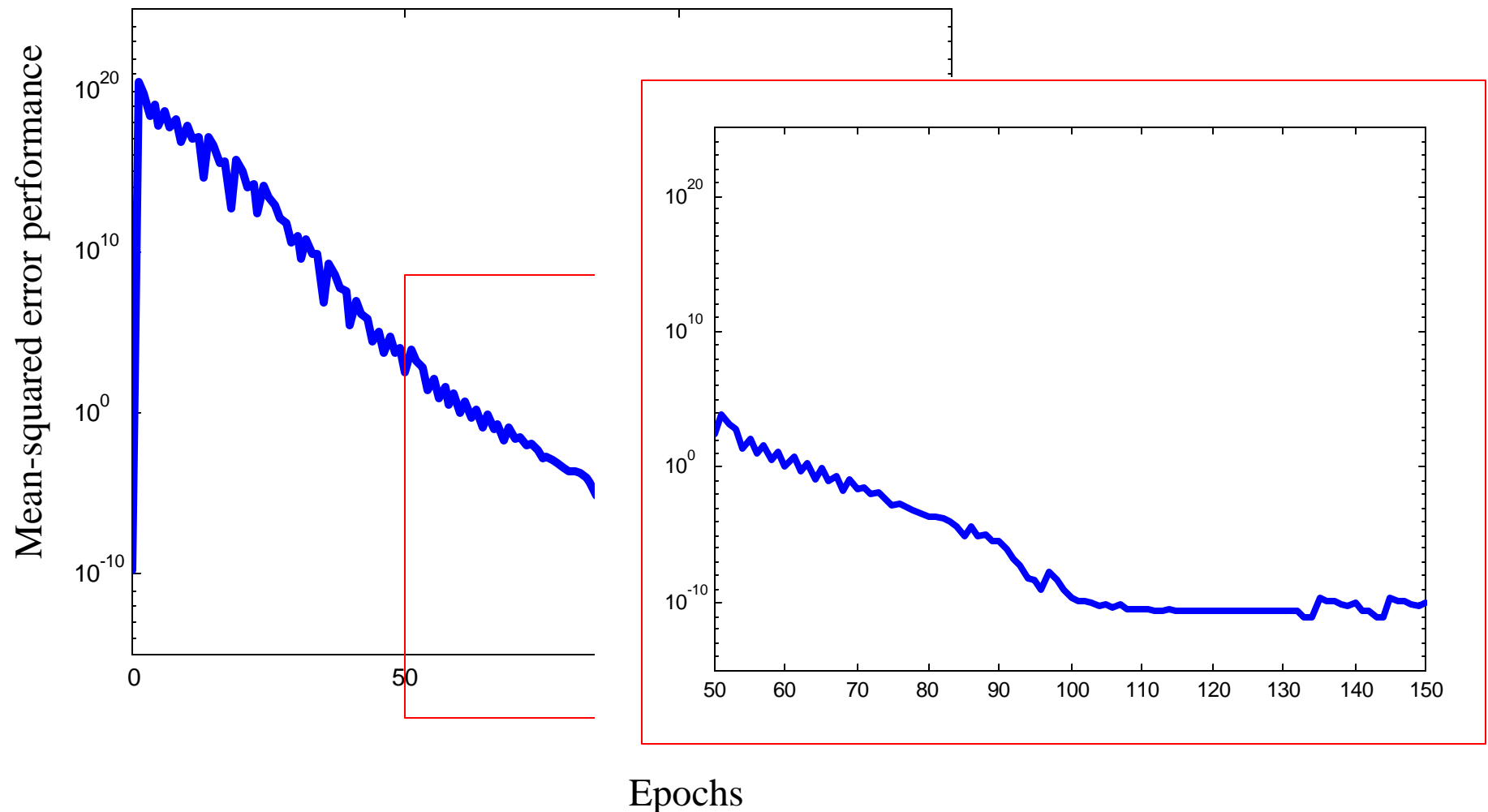
Summary and Conclusions

- Adaptive critic flight controller:

  ❖ Algebraic pre-training based on a-priori knowledge
  ❖ On-line training during simulations (severe conditions)

  Improve aircraft control performance under extreme conditions

- Systematic approach for designing nonlinear control systems, innovative neural network training techniques

- Adaptive critic neural network controller implementation

Future Work:

- Testing: acrobatic maneuvers, severe operating conditions, coupling and nonlinear effects!